

표현형 기반 종분화 진화 신경망을 이용한 다양한 로봇 행동 패턴 생성

(Generating a Variety of Robot Behavior Patterns using Speciation based on Phenotype)

오 근 현 [†] 김 경 중 ^{††}
(Keunhyun Oh) (Kyung-Joong Kim)

조 성 배 ^{†††}
(Sung-Bae Cho)

요 약 환경과 상호작용하는 로봇을 위한 제어를 신경망 진화를 통하여 생성하는 연구가 활발하다. 기존에는 하나의 로봇제어 신경망을 찾는 연구가 주류를 이루었다. 종분화를 이용하여 다양한 행동패턴을 보이는 신경망을 생성하는 연구도 시도되었으나 행동의 특징을 적절히 반영하지 못하는 단점이 있다. 본 논문에서는 로봇이 목적지를 찾아가는 이동경로를 다양화하기 위해 이동 궤적의 특성을 반영한 적합도 공유의 종분화 진화방식을 사용하였다. 형태의 다양성을 위한 유클리드 거리, 폭과 길이의 다양성을 위한 코사인 거리와 아크탄젠트 거리, 위치를 다양화 하기 위한 편집거리를 적합도 공유의 거리 기준으로 사용하여 행동의 특징을 반영하였다. 실험을 통해 로봇의 이동경로를 확인하여 기존 연구에서 사용한 방법보다 더 다양한 행동을 생성한다는 사실을 확인하였다.

키워드 : 로봇 제어기, 진화 로봇공학, 진화 신경망

- 이 논문은 2010년도 교육과학기술부의 재원으로 한국연구재단뇌과학원 천기술 개발사업의 지원을 받아 수행된 연구임(No.2010-0018948)
- 이 논문은 제37회 추계학술발표회에서 '행동 기반 종분화를 이용한 다양한 로봇 제어기 생성'의 제목으로 발표된 논문을 확장한 것임

[†] 학생회원 : 연세대학교 컴퓨터과학과
ocworld@sclab.yonsei.ac.kr

^{††} 정 회 원 : 세종대학교 컴퓨터공학과 교수
kimkj@sejong.ac.kr

^{†††} 종신회원 : 연세대학교 컴퓨터과학과 교수
sbcho@cs.yonsei.ac.kr

논문접수 : 2010년 12월 27일

심사완료 : 2011년 4월 13일

Copyright©2011 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지 : 컴퓨팅의 실제 및 레터 제17권 제5호(2011.5)

Abstract Many people have interests on generating robot controllers interacting with environments by using evolutionary neural networks. Previous studies often aims at finding only one controller. Though some researchers attempted to produce controllers that have various behavioral patterns, they did not use characteristics of behavior. In this paper, we consider features of trajectory of a robot to locate the destination on fitness sharing for speciation. Euclidian distance, cosine and arc tangent distance, and edit distance play a role of representing trajectories. In the experiment, we confirm that the proposed method makes more various behavior patterns than previous studies.

Key words : Robot Controller, Evolutionary Robotics, Evolutionary Neural Networks

1. 서 론

로봇에 대한 관심이 커지면서 모바일 로봇이 환경의 자극에 자율적으로 행동을 결정하는 제어기 생성에 대한 연구가 활발하다. 전문가 설계 방법과 함께 진화 로봇공학(ER)이 널리 사용되고 있다. 진화 로봇공학은 주로 센서를 통해 획득되는 환경의 자극을 입력으로 하고 로봇의 반응인 행동을 출력으로 하는 신경망을 진화시켜 환경과 로봇이 상호작용하는 제어기를 생성하는 방법이다[1].

기존 ER연구는 주로 최적화된 하나의 제어기 탐색을 목적으로 하였다. 이러한 방법은 진화를 통해 학습되는 환경에 과적합된 신경망을 생성해낼 가능성이 있다. 또한 경쟁이나 게임에 로봇을 사용할 경우 단 하나의 전략만이 존재하므로 상대방이 그 특성을 알아내고 대처하기에 수월해진다. 따라서 다양한 패턴을 생성해낼 수 있는 제어기가 필요하다[2]. 이러한 요구에 따라 최근 종분화를 사용하여 다양한 종류의 제어기 생성을 시도하는 연구가 있었으나 행동의 특성을 적절하게 반영하지 못하였다.

본 논문에서는 로봇이 목적지까지 이동하는 경로의 특성을 반영하기 위하여 종분화 진화 신경망을 이용한 다양한 로봇 제어기 생성 방법을 제안한다. 이동경로의 궤적, 폭과 길이, 위치를 종분화 거리 기준으로 설정하여 로봇 행동의 특성을 반영한 다양한 신경망을 생성할 수 있었다.

2. 관련 연구

Knodo는 로봇이 빛을 찾아가는 제어기를 진화 신경망을 통하여 생성하였다. 시뮬레이션이 아닌 현실 환경을 고려한 신경망을 생성하기 위한 방법을 제안하였다. 온라인 학습과 입출력 노드의 최적화를 통하여 진화의

속도를 빠르게 하고 최적의 경로로 이동하도록 하였다 [3]. 로봇의 다양한 행동을 고려하지 않고 하나의 최적화된 신경망을 구하는 연구로 진행되었다.

다양한 패턴을 보이는 제어기를 생성하기 위한 연구는 NEAT와 HyperNEAT를 사용한 방법이 주류를 이루고 있다. 두 방법에서는 신경망 구조기반의 중분화를 사용하였다[4,5]. Drchal 등은 HyperNEAT를 사용하여 로봇이 주어진 경로에서 이동하기 위한 연구를 수행하였다. 생성된 신경망은 서로 구조의 차이는 있지만 문제를 해결하기 위한 다양한 전략이 생성되었는지는 보장할 수가 없다[6].

Trujillo 등은 행동기반 중분화를 수행하기 위해 편집 거리를 사용하였다. 맵을 그리드로 나누어 각 영역마다 문자를 부여하였다. 로봇이 지나간 좌표의 문자를 모아 문자열로 구성하고 신경망들 사이 편집거리를 적합도 공유의 거리 기준으로 삼았다[2]. 이동 경로를 반영하고자 했던 점에서 NEAT보다는 상대적으로 높은 다양성을 보여준다. 하지만 비슷한 이동경로의 궤적이 다른 위치에 나타났을 경우 유사도를 측정할 수 없고, 장애물이 없는 환경에서 행동의 다양성 차이는 크게 나타나지 않았다.

3. 제안하는 방법

그림 1은 로봇 제어기 진화 알고리즘의 개요를 보여준다. p개의 신경망을 생성하여 초기화한다. 그리고 p개의 돌연변이 연산을 적용하여 각각의 신경망에서 변화된 신경망을 한 개씩 생성한다. 각 신경망을 이용하여 로봇이 목적지인 빛을 찾아가도록 한다. 적합도는 신경망이 얼마나 정확히 정해진 목적지에 도착하는지로 측정된다. 적합도 공유를 사용하여 신경망의 적합도를 재평가한다. 이동 경로를 각 시간 별 로봇의 위치들의 특성을 비교하고 부분적으로 비교한 특성의 차이의 합을 통하여 전체 로봇의 이동 경로의 특징을 비교한다. 적합도 공유를 통해 평가된 적합도를 기준으로 가장 높은 값을 가지는 p개의 신경망을 바탕으로 다음 세대의 진화를 수행한다.

3.1 신경망 구조 및 적합도 평가

로봇의 제어기로 신경망을 사용하였다. 빛을 찾아가는 것을 신경망의 목적으로 한다. 주어진 신경망은 실험에 사용된 e-Puck 로봇을 기준으로 설계되었다. 신경망은 2개의 입력과 2개의 출력으로 구성되어 있다. 입력은 로봇으로부터의 빛의 상대적인 방향과 거리를 알 수 있는 IR센서를 사용한다. 출력은 양쪽 바퀴 각각의 속도이다. 양쪽 바퀴의 속도를 조정함으로써 이동 방향과 이동 속도를 조절할 수 있다.

신경망이 빛을 얼마나 정확하게 찾아가는지를 적합도

평가 기준으로 정하였다. 최종 로봇의 목적지와 빛의 가까운 정도를 유클리드 거리를 사용하여 측정하였다.

$$f(i) = \sqrt{(x_{light} - x(last)_i)^2 + (y_{light} - y(last)_i)^2} \quad (1)$$

3.2 로봇 행동 기반 적합도 공유 거리 기준

중분화는 집단의 다양성을 유지하기 위한 방법이다. 본 논문에서는 적합도 공유를 사용하여 중분화하였다. 적합도 공유는 다양성을 유지하기 위해 적합도를 변형시켜주는 방법으로 개체 사이의 거리가 중요한 역할을 한다. 즉 가까운 거리에 있는 개체들은 공유를 많이 하도록 하여 밀집된 개체들이 상대적으로 선택확률이 낮아지도록 하는 방법이다.

$$f'_i = \frac{f_i}{\sum_{j=1}^{\mu} sh(d(i, j))} \quad (2)$$

$$sh(d(i, j)) = \begin{cases} 1 - (d(i, j)/\sigma_{share})^\alpha & \text{if } d(i, j) < \sigma_{share} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

신경망 i 의 현재 적합도를 f_i 라고 하면 공유 적합도 f'_i 은 현재의 적합도를 공유함수 값의 합으로 나누면 된다. μ 는 집단의 크기이고 σ_{share} 는 공유 반경이다. 적합도의 공유는 공유반경 안에 존재하는 개체들 사이에서만 이루어진다. $d(i, j)$ 는 신경망 i 와 신경망 j 사이의 거리를 나타낸다.

유클리드 거리, 코사인 거리, 아크탄젠트 거리, 그리고 편집 거리를 이용하여 로봇 행동의 특성을 구분하였다. 이들을 기반으로 적합도 공유를 이용한 중분화 진화를 수행하였다. 그 결과 표 1과 같은 특성을 보이고 있음을 확인하였다. 이 거리 기준들을 선형 조합하여 사용하면 각 거리 측정 기준의 특성을 반영한 이동경로를 생성할 수 있다.

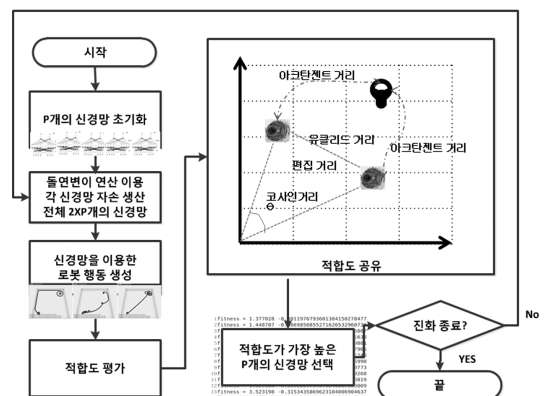


그림 1 제안하는 방법 개요

표 1 거리 기준 별 특성

거리기준	특성
유클리드 거리	다양한 이동경로 형태
코사인 거리	이동경로의 폭과 길이
아크탄젠트 거리	이동경로의 폭과 길이
편집거리	동일한 이동경로의 위치 변화

4개의 거리 기준을 모두 적용한 식은 다음과 같다.

$$d(i, j) = C_1 d(i, j)_{euclidian} + C_2 d(i, j)_{edit} + C_3 d(i, j)_{arctan} + C_4 d(i, j)_{cosine} \quad (4)$$

C_1, C_2, C_3, C_4 의 값은 각 거리 기준의 중요도에 따라서 결정된다. 유클리드 거리가 이동경로 형태의 다양성에 영향을 주기 때문에 가장 중요한 요소로 적용된다. 이 다양화된 경로의 형태의 폭과 길이를 변화하기 위해 코사인 거리와 아크 탄젠트 거리가 사용된다. 코사인 거리의 경우 특정 세대 이상 진화가 진행되면 빛을 찾아가지 못하는 현상이 발생하기 때문에 그 비율을 가장 낮게 잡아야 한다. 편집 거리의 경우 이동경로의 전체적인 위치만이 바뀌는 경우가 많기 때문에 아크탄젠트 거리와의 비율은 임의로 설정해 주어도 된다. 그러므로 각 거리기준의 비율은 $C_1 > C_2 > C_3 > C_4$ 가 된다.

3.3 물리적 거리

적합도 공유에 사용할 수 있는 이동경로의 특성으로 물리적 거리로 변화가 나타나는 궤적의 형태가 있다. 물리적 거리는 신경망이 만들어내는 이동경로의 형태상의 다양성을 유지시켜줄 수 있다. 물리적 거리가 멀지만 목적지를 찾아가기 위해서는 이동경로의 형태가 바뀌어야 하기 때문이다.

$$d(i, j)_{euclidian} = \sum_{n=0}^{max_steps} \sqrt{(x(n)_i - x(n)_j)^2 + (y(n)_i - y(n)_j)^2} \quad (5)$$

신경망 i 와 j 의 유클리드 거리는 로봇이 이동하는 각 단계마다 좌표 사이의 물리적 거리를 비교한 후 각각의 거리를 모두 합하여 구할 수 있다. 유클리드 거리는 이동 경로의 형태는 다양해지지만 각 형태에서 이동경로의 폭이 크게 변하지 않는 특성이 있다.

3.4 각도 거리

적합도 공유시 행동의 각도를 그 특성으로 사용할 수 있다. 행동의 각도는 이동 경로의 폭에 영향을 미친다. 행동의 각도는 두 가지 기준으로 측정할 수 있다. 첫째 각 행동의 시간별 위치를 기준으로 하는 코사인 각도이다. 코사인 값은 0부터 1사이의 값을 가진다. 1에 가까울수록 각도가 0°에 가까우므로 두 포인트의 각도상의 위치는 가까워진다. 신경망 i 와 j 의 코사인 거리는 다음의 식을 이용하여 구할 수 있다.

$$d(i, j)_{cosine} = \sum_{n=0}^{max_steps} (1 - \cos \theta) = \sum_{n=0}^{max_steps} (1 - \frac{A \cdot B}{|A| \cdot |B|}) \quad (6)$$

중분화의 거리 기준으로 쓰이므로 특성의 차이가 크면 거리 또한 크게 해주어야 한다. 따라서 각도의 차이가 많이 날 때의 값의 차이를 크게 하기 위하여 코사인의 최대값인 1에 구해진 코사인 값을 빼주었다. 각 단계의 로봇의 위치 사이의 각도의 차이를 계산한 후 모두 더해주어 전체 이동 경로 사이의 각도의 거리를 구할 수 있다.

두번째 거리 기준으로는 빛과 각 행동의 아크탄젠트 값을 구함으로 알 수 있다. 아크탄젠트 값은 로봇의 해당 시간의 위치가 빛에서 어느 위치에 해당하는지 알 수 있다. 로봇의 위치에서 빛과의 x, y 좌표 차는 다음의 식과 같이 정의한다.

$$\begin{aligned} subY(n)_i &= (y_{light} - y(n)_i) \\ subX(n)_i &= (x_{light} - x(n)_i) \\ subY(n)_j &= (y_{light} - y(n)_j) \\ subX(n)_j &= (x_{light} - x(n)_j) \end{aligned} \quad (7)$$

n 은 로봇이 이동할 때의 단계, 즉 시간을 의미한다. 빛의 위치는 (x_{light}, y_{light}) 로 정의하고 n 단계에서 신경망 i 와 j 로 인해 로봇이 도착한 위치는 각각 $(x(n)_i, y(n)_i)$ 와 $(x(n)_j, y(n)_j)$ 로 정의한다. 식 (7)을 이용하여 아크탄젠트에 의한 거리 차를 구하는 식은 다음과 같다.

$$d(i, j)_{arctan} = \sum_{n=0}^{max_steps} |arctan(subY(n)_i/subX(n)_i) - arctan(subY(n)_j/subX(n)_j)| \quad (8)$$

신경망 i 와 j 사이의 아크탄젠트 값은 i 신경망으로 이동한 로봇의 n 시점에서의 위치와 빛 사이의 아크 탄젠트 값과 j 신경망으로 이동한 로봇의 n 단계에서 빛 사이의 아크탄젠트 값의 차의 절대값을 구한다. 이 값의 크기가 작을수록 두 점이 빛을 기준으로 가까운 각도상에 위치하는 것을 의미한다. 즉, 빛을 기준으로 각도 값의 특성이 유사하다는 것을 의미한다.

각도의 특성들은 로봇의 이동경로의 형태를 다양하게 생성하지는 못하지만 동일한 형태의 이동경로의 폭과 거리를 변하게 해주는 특성을 가지고 있다. 코사인 거리의 경우 특정 세대 이상으로 진화를 하면 로봇이 빛을 찾지 못하게 되는데 그 이유는 각도 값으로 비교를 하였기 때문에, 출발선상에 가까운 곳에서만 이동하여도 각도를 기준으로 한 거리가 결정되기 때문이다.

3.5 편집거리

동일한 이동 경로를 피하기 위해 편집거리를 사용하여 신경망들 사이의 거리를 측정하여 적합도 공유의 거리 기준으로 사용할 수 있다. 편집거리란 두 개의 문자

```

int editDistance
(char s[1..m], char t[1..n]){
  declare int d[0..m, 0..n]
  for i from 0 to m
    d[i, 0] := i // 삭제
  for j from 0 to n
    d[0, j] := j // 삽입
  for j from 1 to n {
    for i from 1 to m {
      if s[i] = t[j] then
        d[i, j] := d[i-1, j-1]
      else
        d[i, j] := minimum (
          d[i-1, j] + 1, // 삭제
          d[i, j-1] + 1, // 삽입
          d[i-1, j-1] + 1 // 대체
        )
    }
  }
  return d[m,n]
}

```

그림 2 편집거리 의사 코드

열이나 수열이 주어졌을 때, 원본 수열을 원하는 수열로 바꾸는 최소 편집 횟수를 구하는 알고리즘이다. 즉, 두 문자열 사이의 편집거리는 하나의 문자열을 다른 하나의 문자열로 변환하기 위해 요구되는 연산의 수이다. 그림 2는 편집거리의 의사 코드를 나타낸다. 입력은 비교할 두 문자열 s 와 t 이고 출력은 계산된 편집거리이다.

전체 맵을 격자 형태로 나누어서 각 영역마다 특정 문자를 부여한다. 그리고 로봇이 특정 영역을 지나갈 때 문자열에 그 영역의 문자를 추가한다. 이러한 과정의 예시를 그림 3에서 나타내었다. 전체 맵을 4×4 그리드로 나누었고 각 부분에 고유한 문자를 부여하였다. 빗금표시가 있는 부분이 로봇이 이동한 경로이다. 이 경로에 따라서 문자열 S 의 값이 결정 된다. 이렇게 각 신경망의 결과로 나온 로봇의 이동 경로들을 편집거리 값을 이용하여 거리 기준을 측정한다. 편집거리를 기준으로 적합도 공유로 중분화 진화 연산을 수행하였을 경우 행동 경로에 큰 폭의 차이는 없다. 하지만 같은 형태의 이동경로가 다른 위치에서 나타나는 모습을 확인할 수 있다.

A1	A2	A3	A4
B1	B2	B3	B4
C1	C2	C3	C4
D1	D2	D3	D4

$$S = D_1 C_2 B_3 A_3 A_4$$

그림 3 격자 형태로 나눈 맵을 문자열로 표현한 예



그림 4 e-Puck 로봇

4. 실험 및 평가

4.1 환경

실험은 Enki시뮬레이터(<http://home.gna.org/enki/>) 환경에서 진행하였다. 대상 로봇으로 e-puck(<http://www.e-puck.org/>)을 사용하였다. 맵의 크기는 가로, 세로 120cm로 정의하였다. 좌표는 1cm단위로 계산하며 (x,y) 형태로 표기한다. 빛의 위치는 $(100,100)$ 이고 로봇의 출발 위치는 $(20,20)$ 으로 설정하였다.

그림 1에서 신경망의 수를 나타내는 p 는 20으로 정하였다. 제안하는 적합도 공유 거리 기준의 값은 $C_1 = 10$, $C_2 = 2$, $C_3 = 1$, $C_4 = 1/2$ 로 설정하였다. 3절에서 언급하였던 물리적 거리, 각도 거리, 편집거리 각각에 최대 가중치를 부여하여 실험을 수행하여 각 중분화 거리 기준의 특징을 확인하였다. 또한 제안하는 방법에서 생성된 다양한 이동경로를 확인하여 각 거리 기준의 특징들을 모두 반영함을 확인하였다. 진화는 각각 1000세대씩 수행하였다.

4.2 평가

20대 중후반인 3명의 대학원생이 함께 보고 나는 행동의 개수와 각 거리 기준별 특성들을 파악하였다. 표 2는 실험 결과를 정리한 내용이다. 각도 거리에 가장 큰 가중치를 부여하였을 경우 이동 궤적의 다양성은 크게 나타나지 않으나 각 구간별 폭과 길이에 영향을 끼침을 확인하였다. 기존 연구에서 사용한 편집거리 위주의 실험에서는 장애물이 없는 환경상에서는 종의 차이가 나타나지 않는 것을 확인하였다. 그림 5에서 볼 수 있듯이, 궤적이 큰 차이가 없음을 확인할 수 있다. 제안하는 방법과의 비교를 위한 유클리드 거리만 사용한 실험에

표 2 실험 결과

거리기준	중수	특징
각도 거리 위주	2	궤적 길이 너무 변화
편집 거리 위주	1	기존 연구에서 사용, 궤적 위치 변화
유클리드 거리만	4	궤적의 모양 변화
제안하는 방법	8	각 거리기준 특징 반영

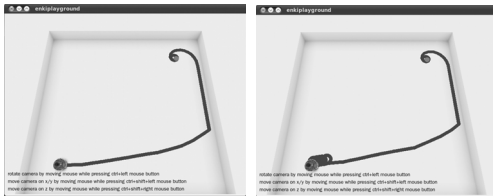


그림 5 편집거리(기존연구)기반 실험 결과

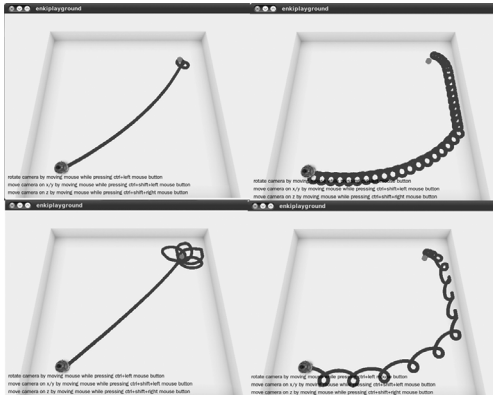


그림 6 유클리드 거리기반 실험 결과

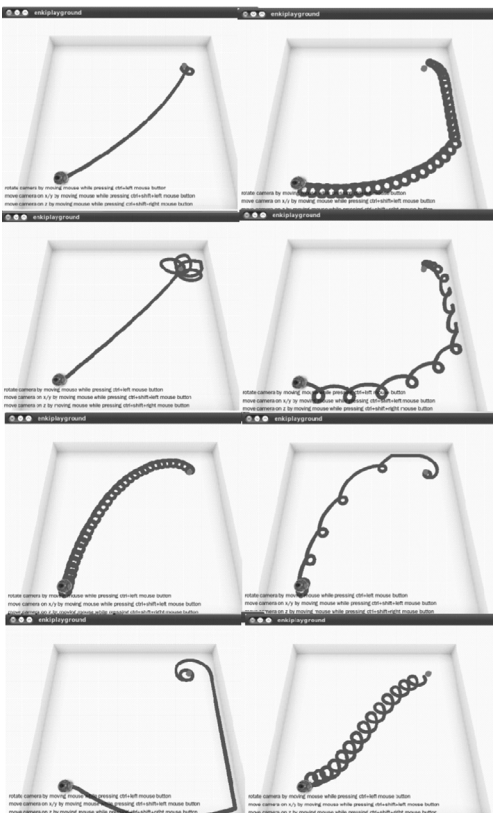


그림 7 제안하는 방법의 실험 결과

서는 그림 6과 같이 꺾적 모양의 다양성이 나타났다. 제안하는 방법의 순으로 가중치를 부여하였을 때 그림 7과 같이 각 거리기준의 특성을 반영하여 유클리드 거리만 사용하였을 경우보다 더 다양한 종이 분화됨을 확인하였다.

5. 결 론

본 논문에서는 행동 기반의 종분화 진화 신경망을 통한 로봇 제어기 생성 방법을 제안하였다. 목적지를 찾아가는 로봇의 행동인 이동경로의 특징을 물리적 거리, 각도 거리, 편집 거리로 구분하였다. 이 특징들을 기반으로 거리 기준을 설정하였다. 이동경로형태의 다양성을 위해 물리적 거리를 측정하는 유클리드 거리에 가장 높은 가중치를 주고 다음으로 각도 기준 중 하나인 아크 탄젠트 거리와 이동 영역에 기반한 편집거리, 마지막으로 코사인 거리 순으로 가중치를 부여하여 이들을 함께 사용하였을 경우 각 거리 기준들의 특성을 모두 반영한 다양한 이동경로들이 생성됨을 실험을 통해 확인할 수 있었다.

공간상의 물리적 장애물과 다수의 로봇이 함께 목적지를 찾아가는 행위의 다양성에 대한 연구를 향후 진행할 예정이다.

참 고 문 헌

- [1] S. Nolfi, and D. Floreano, "Evolutionary robotics: The biology, intelligence, and technology of self-organizing machines," Bradford Book, 2004.
- [2] L. Trujillo, E. Lutton, and F. F. Vega, "Behavior-based speciation for evolutionary robotics," *Proc. of Genetic and Evolutionary Computation Conference*, pp.297-298, 2008.
- [3] T. Knodo, "Evolutionary design and behavior analysis of neuromodulatory neural networks for mobile robots control," *Applied Soft Computing*, vol.7, no.1, pp.189-202, 2007.
- [4] K. O. Stanley, and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evolutionary Computation*, vol.10, no.2, pp.99-127, 2002.
- [5] K. O. Stanley, and D. B. D'Ambrosio, and J. Gauci, "A hypercube-based encoding for evolving large-scale neural networks," *Artificial Life*, vol.15, no.2, pp.185-212, 2009.
- [6] J. Drchal, J. Koutnik, and M. Snorek, "HyperNEAT cotrolled robots learn how to drive on roads in simulated environment," *Proc. of Congress on Evolutionary Computation*, pp.1097-1092, 2009.